

IN THE CLAIMS:

This listing of claims will replace all prior versions, and listings of claims in the application:

Claims 1-58 (cancelled).

Listing of Claims:

59. (Amended) A system for exchanging data, comprising
- a) a third party server;
 - b) a web host server;
 - c) a commerce server having a trading partner profile table;
 - d) a first network connecting the customer computer, web host server and commerce server;
 - e) an applications server connected to the commerce server by a second network, the applications server responsive to remote procedure calls from the commerce server.
60. (previously presented) The system for exchanging data as in claim 59, wherein the first network comprises a plurality of segments.
61. (previously presented) The system for exchanging data as in claim 60, wherein at least one segment of the first network is selected from the group consisting of wireless, fiber optic, infrared, a hand held computer, and a voice recognition device.
62. (previously presented) The system for exchanging data as in claim 59, wherein the second network comprises a plurality of segments.
63. (previously presented) The system for exchanging data as in claim 62, wherein at least one segment of the second network is selected from the group consisting of wireless, fiber optic, infrared, a hand held computer, and a voice recognition device.
64. (previously presented) The system for exchanging data as in claim 63, wherein the initiator further comprises a web browser.

65. (previously presented) A method for exchanging data, comprising;
- a) an initiator who initiates the transaction, the transaction including data, selected from the group consisting of an application server, a third party server, a web host server, and a commerce server;
 - b) a responder which receives the transaction selected from the group consisting of the application server, the third party server, the web host server, and the commerce server;
 - c) a point to point secure transfer protocol using high level encryption for sending and receiving the transaction, the protocol comprising;
 - 1) computer readable instruction code means for establishing an active listener via an event wait state;
 - 2) computer readable instruction code means for accessing the trading partner profile table and determining the identity of the initiator, what transactions the initiator and responder have mutually agreed to allow, determine a location and format of data for the transaction and determine allowable values;
 - 3) computer readable instruction code means for generating a security error and terminating the code if the initiator is not authorized;
 - 4) computer readable instruction code means for writing activity to an activity log;
 - 5) computer readable instruction code means for determining and processing an event state, the event state selected from the group consisting of idle, session request, session confirm, key request, key confirm, data package, next data package, package confirm, end request, and end confirm;
 - 6) establishing a business conversation between trading partners, the business conversation comprised of specific time or event driven transaction sets;
 - 7) computer readable instruction code means for building a header and cargo appropriate for the event state;
 - 8) computer readable instruction code means for generating a unique encryption key pair for each transmission;

- 9) computer readable instructions for compressing and encrypting the data using the unique encryption key pair;
- 10) computer readable instruction means for sending the data to the responder that prevents the data from being stored on a server hard drive while the data is in transit between the initiator and responder;
- 11) computer readable instruction code means for receiving, decrypting and decompressing the data.

66. (previously presented) The method for exchanging data as in claim 65, wherein the data comprises at least one of the group consisting of text., binary objects, image, a sound recording, a data stream, EDI, XML, and EDIFACT.

67. (previously presented) The method for exchanging data as in claim 65, wherein a unique signature key generated on the hosting system is derived from a passphrase generated from user input and unique system identifiers facilitating non-repudiation.

68. (previously presented) The method for exchanging data as in claim 65, wherein sharing of public keys is directly between trading partners only and used during a single session only.

69. (previously presented) The method for exchanging data as in claim 65, wherein the initiator's and responder's public keys are uniquely created by the insertion of string values into randomly chosen positions.

70. (previously presented) The method for exchanging data as in claim 65, wherein bi-directional verification of sender and recipient identities is accomplished prior to any exchange of data.

71. (previously presented) The method for exchanging data as in claim 65, wherein separate exchanges of public signature keys, used for trading partner validation, and public exchange keys, used for encoding/decoding of data, are facilitated.

72. (previously presented) The method for exchanging data as in claim 65, wherein the initiator maintains full control of data provided to validated partners.

73. (previously presented) The method for exchanging data as in claim 65, wherein an entire data package is encoded prior to transmission.

74. (previously presented) The method for exchanging data as in claim 65, wherein data receipt by the intended recipient is verified.

75. (previously presented) A queue based method for initiating and managing a business conversation, wherein an inbound transmission originates from the group consisting of a web host and a third party server, the business conversation comprising at least one transaction and comprising a commerce server, the commerce server comprising a trading partner profile table, a transaction engine queue, a reply requirements queue, a transaction engine outbound queue, a SDS transaction queue, and a transport protocol outbound queue, the commerce server being communicatively coupled by a first network to at least one of a web host server and a third party server, and the commerce server being communicatively coupled by a second network to an application server, the steps comprising:

- a) receiving an inbound request, determining the initiator and responder, decode and decompress the request, determine the output destination and adding to the transaction engine queue;
- b) parsing the inbound transmission into at least one transaction, authorizing the initiator, preparing a data structure for each transaction and returning the transaction to the transaction engine queue;
- c) building and managing the business conversation by utilizing the business transaction map and forwarding the transactions to the appropriate queue selected from the reply requirements queue, the transport protocol outbound queue and the SDS transaction queue;

76. (previously presented) The queue based method for initiating a business conversation according to claim 75 wherein the inbound transmission is received from the group consisting of the web host server and third party server communicatively coupled to the first network.

77. (previously presented) The queue based method for initiating a business conversation according to claim 75 wherein the final destination of the transaction is the application server.

78. (previously presented) The queue based method for initiating a business conversation according to claim 75 wherein step (c) further comprises building and managing at least one future conversational transaction and forwarding the future transaction to the reply requirements queue.

79. (previously presented) The queue based method for initiating a business conversation according to claim 75 wherein step (c) further comprises sending an outbound transaction to the transport protocol outbound queue.

80. (previously presented) The queue based method for initiating a business conversation according to claim 75 wherein step (c) further comprises sending the transaction from the SDS transaction queue to its final destination, the application server.

81. (previously presented) A queue based method for initiating and managing a business conversation, wherein an outbound transmission originates from an applications server, the business conversation comprising at least one transaction and comprising a commerce server, the commerce server comprising a trading partner profile table, a transaction engine queue, a reply requirements queue, a transaction engine outbound queue, a SDS transaction queue, and a transport protocol outbound queue, the commerce server being communicatively coupled by a first network to at least one of a web host server and a third party server, and the commerce server being communicatively coupled by a second network to an application server, the steps comprising:

- a) receiving an outbound transaction, reading, parsing and formatting of the transaction and adding to the transaction engine outbound queue;
- b) parsing of transaction to the recipients pre-determined format based on the overall database structure map;
- c) determining the recipient of at least one transaction, building the outbound transaction and forwarding to the transaction engine queue;
- d) building and managing of at least one future conversational transaction and forwarding the future transaction to the reply requirements queue;
- e) compressing, logging and encoding the outbound transaction;
- f) transmission of the transaction to it's final destination.

82. (previously presented) The queue based method for sending an outbound transmission of a business conversation according to claim 81 wherein the final destination of the outbound transmission is selected from the group of the web host server and third party server on the first network.